

# Oracle NoSQL Database als weiteres Standbein

**Frank Püchl**  
**PAYBACK GmbH**  
**München**

**Co-Author:**  
**Carsten Czarski, ORACLE Deutschland B.V. & Co KG, München**

## **Schlüsselworte**

Datenbank, NoSQL, Key-Value-Store, closed-loop, operationales Warehouse, Praxisbeispiel

## **Einleitung**

PAYBACK nutzt die Oracle NoSQL DB als "Low Latency" System für das zentrale Coupon-Management. Coupons und deren Zuordnung zu PAYBACK-Mitgliedern sowie Informationen zur Segmentierung des PAYBACK-Mitglieds werden innerhalb der Oracle NoSQL DB gespeichert. Dadurch konnte die Last auf den bestehenden Data Warehouse und OLTP Systemen signifikant reduziert werden.

Der Vortrag stellt die Oracle NoSQL DB, deren Unterschiede zum bekannten RDBMS Oracle und deren Besonderheiten im Vergleich zu anderen NoSQL-Datenbanken vor und zeigt auf, wie die Oracle NoSQL DB bei PAYBACK aufgesetzt und mit den bestehenden Systemen integriert wurde. Konkrete Projekterfahrungen, Lessons learned und ein Ausblick auf die Zukunft runden den Vortrag ab.

## **Über PAYBACK**

PAYBACK ist einer größten Marketing- und Couponingplattformen weltweit. In Deutschland können Kunden mit nur einer Karte bei über 600 im Alltag relevanten Unternehmen im Handel, online und mobil Punkte sammeln und diese gegen Gutscheine, Prämien oder Lufthansa-Meilen einlösen bzw. spenden. Im Jahr 2012 betrug der Wert der gesammelten Punkte 174 Mio. EUR, 95 Prozent davon wurden von den Kunden wieder eingelöst. Als Multichannel-Marketingplattform und „Triologue Company“ kommuniziert PAYBACK synchron über alle Kanäle: Offline, online und mobil – und verbindet dabei Reichweite optimal mit Personalisierung. Unternehmen haben die Möglichkeit, integrierte Kampagnen über alle verfügbaren Kanäle zu betreiben. Sie können ihre Kunden zum richtigen Zeitpunkt am richtigen Ort mit den relevanten Botschaften ansprechen und erzielen dadurch den größtmöglichen ROI. International sammeln über 50 Millionen Kunden Punkte mit PAYBACK. PAYBACK ist eine Tochtergesellschaft der Management Holding Loyalty Partner GmbH, dieser wiederum ist Teil der American Express Group.

## **Personalisierung**

Die Personalisierung erfolgt auf historischen Daten, wird im Data Warehouse errechnet und anschließend zurück in das OLTP-System gespielt. Hierbei erfolgt der Austausch durch ein Flatfile, um die Systeme zu entkoppeln. Abhängig von den zur Personalisierung verwendeten Attributen können hierbei Dateien mit Größen im zweistelligen Gigabyte-Bereich entstehen.

Die Daten werden im OLTP (RDBMS) in einer Tabelle abgelegt. Da die Aktualisierungen jeweils mehr als 30% aller Datensätze betreffen, wird die Tabelle immer vollständig neu geladen. Um Verluste von Services zu vermeiden, existieren zwei gleich aufgebaute Tabellen. Eine der beiden Tabellen wird per Synonym der Applikation lesend zur Verfügung gestellt, die andere Tabelle wird zur Neuladung per SQL\* Loader und anschließendem Neuaufbau des Index verwendet. Nach

erfolgreicher Ladung wird das Synonym so geändert, dass es auf die Tabelle mit den neuen Daten zeigt.

Beim Laden der Daten in das OLTP-System entsteht sehr viel IO-Last, welche sich negativ auf das Gesamtsystem auswirken kann. Das Muster für Befüllung und Abfrage erfordert nicht zwingend ein RDBMS.

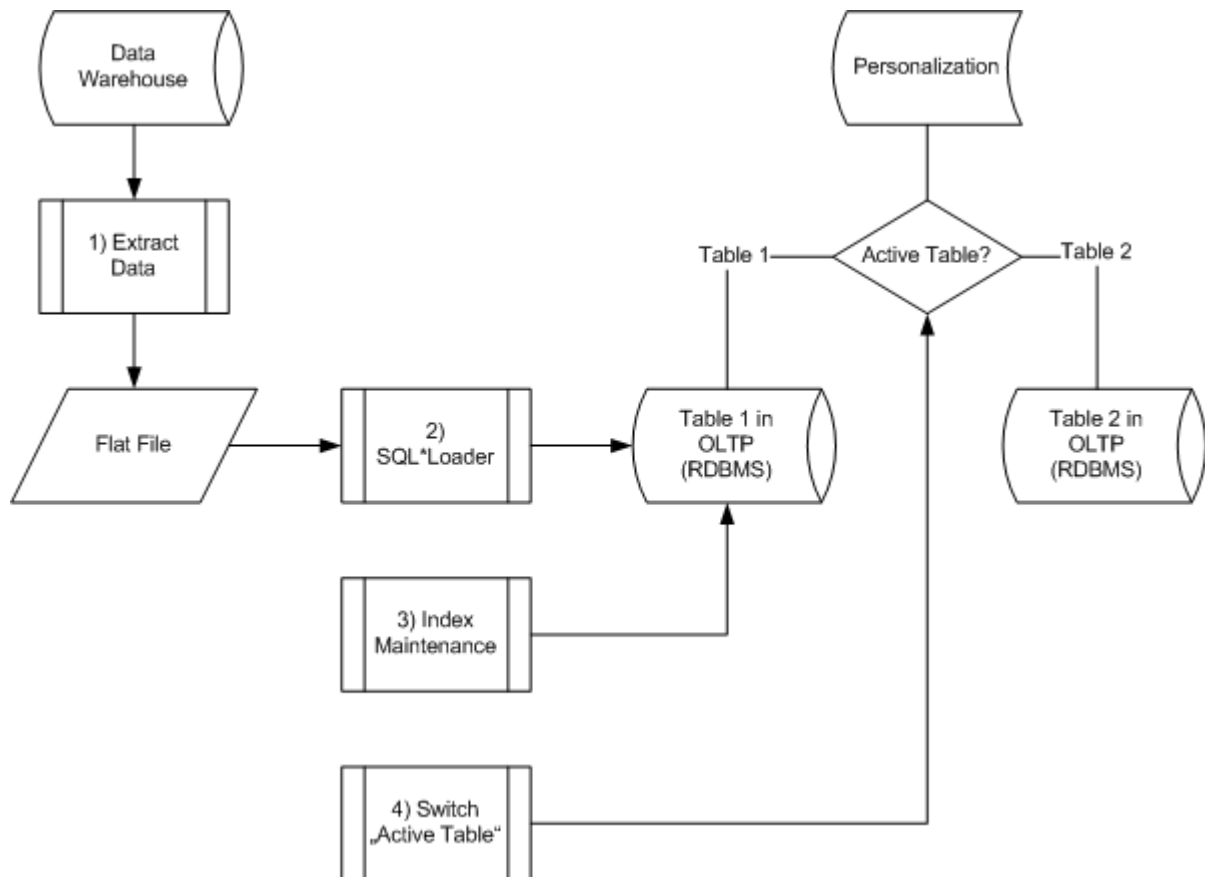


Abb. 1: Beladung des OLTP unter RDBMS Oracle

### Oracle NoSQL Database

Zur Markteinführung von PAYBACK in Mexiko im Jahre 2012 wurde ein neuer Weg begangen. Die Personalisierung wurde aus der relationalen Datenbank entfernt und in einen Key-Value-Store transformiert. Möglich wurde dies, da der Zugriff auf die Daten rein Key-basiert erfolgt. Gleichzeitig konnte hierbei von noch mehr Flexibilität zur Speicherung der Attribute gewonnen werden.

PAYBACK hat sich für die zum damaligen Zeitpunkt neue Oracle NoSQL Database entschieden, um weiterhin einen renommierten, zuverlässigen und starken Partner für das Datenmanagement zu haben. Neben der kostenpflichtigen Enterprise Edition wird auch eine kostenfreie Community Edition angeboten, welche zum Aufbau der lokalen Entwicklungsumgebung bei weitestgehend gleicher Funktionalität verwendet werden kann.

Der bisherige Extrakt der Daten aus dem Data Warehouse bleibt unangetastet. Anstelle der Tabellen im OLTP-System werden die Daten in dem Key-Value-Store (KVStore) der Oracle NoSQL DB gespeichert. Die Aktualisierung erfolgt hierbei inkrementell, bei der Aktualisierung kann nun Teil-

oder Vollbeladung über genau denselben Ladeprozess erfolgen. Der KVStore bietet optimale Performance und erlaubt über Consistency und Durability weitere Tuningmöglichkeiten.

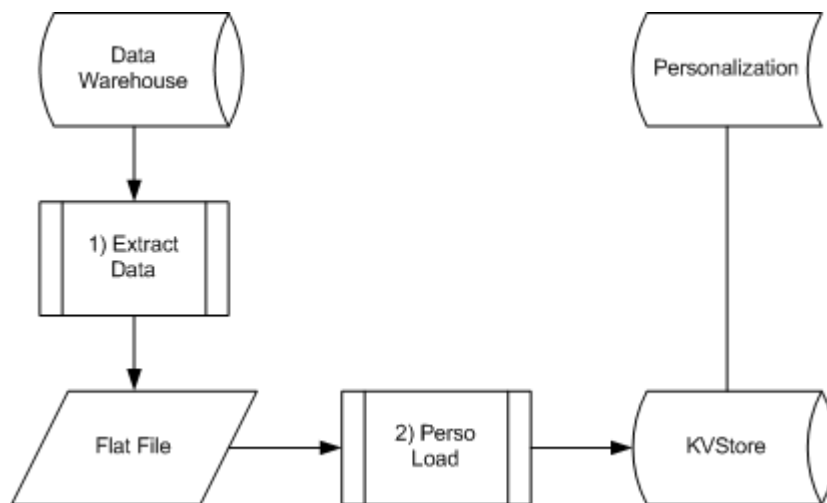


Abb. 2: Beladung des OLTP unter Oracle NoSQL DB

### Profil der Oracle NoSQL-Datenbank

Die Oracle NoSQL Datenbank wurde auf Basis der schon vor längerer Zeit von Oracle akquirierten BerkeleyDB implementiert. Wie bereits angedeutet, ist sie ein Key-Value-Store mit zusätzlichen Mechanismen zur verteilten Datenhaltung: Die gespeicherten Daten müssen auf verschiedene Rechnerknoten verteilt werden, es muss eine Replikation stattfinden, und das System muss mit dem Ausfall einzelner Teile fertig werden.

Die Oracle NoSQL DB ist als Community Edition (CE) oder als Enterprise Edition (EE) erhältlich. Die Community Edition ist kostenlos und eine OpenSource Lizenz. Die Enterprise Edition ist eine kommerzielle Lizenz mit der Möglichkeit eines Wartungsvertrags. Features der Enterprise Edition beziehen sich zur Zeit vor allem auf die Integration in ein Rechenzentrum: So gehört das JMX-Monitoring oder die Möglichkeit, Daten der NoSQL DB als externe Tabelle im RDBMS verfügbar zu machen, zur Enterprise Edition.

Da es keine Abfragesprache für die NoSQL-Datenbank gibt („No SQL“), greift die Java-Anwendung mit Programmier-Aufrufen (API) zu. Die Kommunikation übernimmt dabei der Oracle NoSQL DB Treiber. Die bereitgestellten Zugriffsverfahren sind sehr einfach und beschränken sich im wesentlichen auf **GET**, **PUT** und **DELETE**-Operationen. Das folgende Listing zeigt, wie ein Key-Value-Paar in der Oracle NoSQL DB gespeichert wird.

```
import oracle.kv.*;

public class NoSQLDB_Store {
    public static void main(String args[]) {
        KVStore store = KVStoreFactory.getStore (
            new KVStoreConfig(
                "MyKVStore",
                new String[] {"storagenode001:5000", "storagenode002:5000"}
            )
        );

        store.put(
            Key.createKey("DiesIstEinKey"),
            Value.createValue(new String("Dies ist ein Wert").getBytes())
        );

        store.close();
    }
}
```

*Listing: Beispielcode zum Speichern eines Key-Value Paares in der Oracle NoSQL DB*

Die NoSQL-Datenbank wird auf einem Cluster mit Storage Nodes installiert. Ein Storage Node ist ein physikalischer Rechnerknoten mit lokalen Plattensystemen, auf dem die NoSQL-Datenbanksoftware läuft. Alle Storage Nodes speichern ihre Daten lokal. Zur Verteilung der Daten auf die Storage Nodes werden die Schlüssel auf Partitionen abgebildet. Beim Aufsetzen der NoSQL DB-Topologie entscheidet man sich für die Anzahl Partitionen, die man verwenden möchte. Die Key-Value-Paare werden beim Schreiben per Hash-Algorithmus gleichmäßig über alle Partitionen verteilt. Die Partitionen selbst werden auf verschiedene Storage Nodes verteilt, wodurch sich das verteilte System ergibt (Abb. 3).

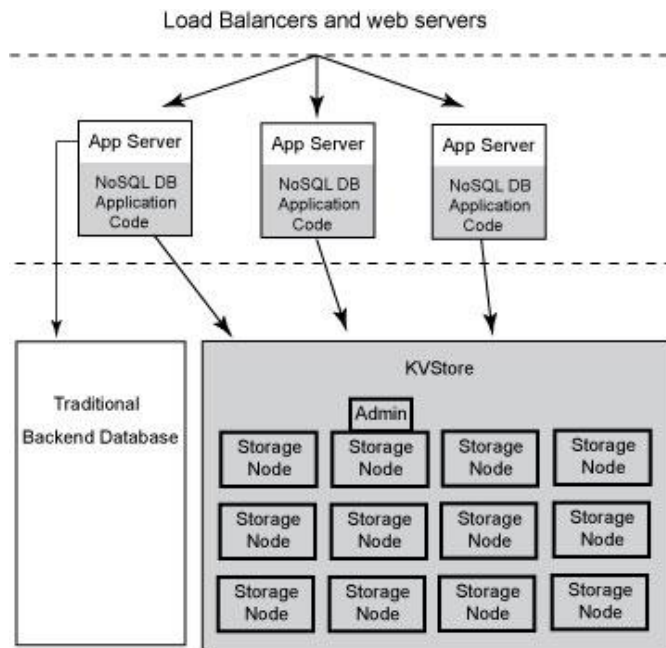


Abb. 3: Topologie eines NoSQL-Datenbanksystems

Damit das Gesamtsystem auch bei Ausfall eines Storage Nodes verfügbar bleibt, müssen die Daten repliziert werden. Eine Replikationsgruppe besteht aus einem Master und mindestens einer Replika und ist für die redundante Speicherung einer bis mehrerer Partitionen verantwortlich – empfohlen ist allerdings genau eine Partition pro Replikationsgruppe. Der Replikationsfaktor ist definiert als „Master + Anzahl Replikas“. Ein Master und zwei Replikas bedeuten also einen Replikationsfaktor von 3. Werden Daten in der NoSQL DB gespeichert, so geschieht dies immer zunächst auf dem Master – anschließend wird repliziert. Leseanfragen können, je nach der Einstellung zur Lesekonsistenz, entweder vom Master oder von einer der Replikas bedient werden.

Fällt ein Master aus, so wählen die verbliebenen Knoten der Replikationsgruppe einen neuen – aus diesem Grund ist ein Replikationsfaktor von wenigstens 3 zu empfehlen, nur dann sind bei Ausfall des Masters noch zwei Knoten übrig, von denen einer der neue Master werden kann. Die eingerichtete Topologie kann auch im laufenden Betrieb verändert werden. Neben dem einfachen Hinzufügen oder Entfernen von Storage Nodes ist auch das Hinterlegen einer komplett neuen Topologie möglich. Die NoSQL DB verteilt die Daten dann entsprechend neu.

### Projekterfahrung

Der sogenannte KV Lite Store erlaubt den einfachen Einsatz der Oracle NoSQL DB auf den lokalen Entwicklungsumgebungen. Hierbei wird ein Single-Node Store mit einer Replikationsgruppe ohne Installationsaufwand bereitgestellt.

Der erste Produktionseinsatz erfolgt mit Version 1.2.123 der Oracle NoSQL DB. Die fehlende Einschränkung der Konfiguration von Port Ranges für die interne Kommunikation sowie einige Kinderkrankheiten werden in der aktuellen Version 2.1.8 mit der offiziellen Bezeichnung „Oracle NoSQL Database 12cR1 (12.1.2.1.8)“ behoben.

Weiterhin bestehen bleibt allerdings die fehlende Authentifizierung/Autorisierung was durch entsprechende Maßnahmen im Netzwerk unbedingt zu berücksichtigen ist.

Zusammen mit dem Upgrade führt PAYBACK das, ab Version 2.0 unterstützte, Avro-Format zur Speicherung des Values ein. Dieses Format unterstützt, durch Zuhilfenahme von Schema-Definitionen, die Kompatibilität unterschiedlicher Versionen eines Schemas. Aufwändige Datenmigrationen gehören damit der Vergangenheit an.

Als besonderes „Schmankerl“ wird in der aktuellen Version 2.1.8 auch ein Data Command Line Interface zur Verfügung gestellt. Zusammen mit dem Avro-Format steht hiermit eine einfache Möglichkeit für den Zugriff auf den KVStore zur Verfügung.

### **Ausblick**

Die Nutzung der Oracle NoSQL DB für überwiegende Lesezugriffe rechtfertigt den Aufwand einer zusätzlichen Datenbank nicht. Wir setzen die Oracle NoSQL DB für weitere Themen ein um auch die Vorteile der schnellen Schreibzugriffe nutzen zu können.

Ein wichtiges Feature von PAYBACK ist das sofortige Sammeln von Punkten ohne aufwändige Registrierung zum Erhalt einer PAYBACK Karte. Zur vollen Nutzung von PAYBACK ist jedoch eine nachträgliche Registrierung zwingend erforderlich. Zur Absicherung des Systems gegen betrügerische Registrierungen werden Anmeldeversuche in der Oracle NoSQL DB gespeichert und können somit ausgewertet und zur Realtime Fraud Detection herangezogen werden.

Ein weiteres Einsatzgebiet der Oracle NoSQL DB ist die Speicherung der für OAuth 2.0 benötigten Informationen. Die (derzeit noch) fehlenden Sekundärindexe werden durch entsprechende zusätzliche Key-Value Einträge kompensiert.

Der dritte, sicherlich jedoch nicht der letzte, Anwendungsfall bildet die kurzfristige Speicherung von Authentifizierungsschlüsseln wie man diese bei Bestätigungen von Email-Adressen oder dem Zurücksetzen von Kennwörtern kennt,

### **Kontaktadresse:**

Frank Püchl  
PAYBACK GmbH  
Theresienhöhe 12  
D-80339 München

Telefon: +49 (0) 89-99741 323  
E-Mail frank.puechl@payback.net  
Internet: www.payback.net